[Partly ghostwritten from her draft]

HyperLab as a Human Media System

Marlene Mallicoat, Research Associate, Sapporo HyperLab

The Sapporo HyperLab was set up for the study and creative development of computer media. However, it has also proven to be a laboratory for the study and creative development of another kind of information medium-- human interaction.

The HyperLab has brought together the Japanese spirit of teamwork and cooperation and the American style of individualistic and imaginative energy.

I believe the HyperLab is flourishing due to the merging of the best qualities of both cultures, and of the skills and interests of its varied members.

The Lab Environment

There are many kinds of human interaction to study in the Lab-- many subjects, tasks, puzzles, surprises, questions, explanations, investigations, jokes, conversations-- from Apples to Zips, Ethernet and Internet, Digicash and software crash. Deep discussions among team members are punctuated by much laughter and light-heartedness.

It's not easy to achieve the kind of harmony and productivity that characterize the Lab, but that harmony has been naturally easier in the spacious and pleasant surroundings that have been so generously provided-- a cheery and well-managed setting, thoughtfully watched over by SEC and our able office manager.

There are many long hours at the screen

The working day sometimes lasts from lunch to dawn ("It's a real Lab now!" said Ted, when some of the other members started working until dawn along with him.) We are grateful for the all-night 7-11 store downstairs, and the non-smoking environment which the members so considerately maintain.

HyperLab
as a Human Media System

Marlene Mallicoat, Research Associate, Sapporo HyperLab

The Sapporo HyperLab was set up for the study and creative development of computer media. However, it has also proven to be a laboratory for the study and creative development of another kind of information medium-- human interaction.

The HyperLab has brought together the Japanese spirit of teamwork and cooperation and the American style of individualistic and imaginative energy.

I believe the HyperLab is flourishing due to the merging of the best qualities of both cultures, and of the skills and interests of its varied members.

The Lab Environment

There are many kinds of human interaction to study in the Lab-- many subjects, tasks, puzzles, surprises, questions, explanations, investigations, jokes, conversations-- from Apples to Zips, Ethernet and Internet, Digicash and software crash. Deep discussions among team members are punctuated by much laughter and lighthearedness.

It's not easy to achieve the kind of harmony and productivity that characterize the Lab, but that harmony has been naturally easier in the spacious and pleasant surroundings that have been so generously provided-- a cheery and well-managed setting, thoughtfully watched over by SEC and our able office manager.

There are many long hours at the screen

The working day sometimes lasts from lunch to dawn ("It's a real Lab now!" said Ted, when some of the other members started working until dawn along with him.) We are grateful for the all-night 7-11 store downstairs, and the non-smoking environment which the members so considerately maintain.

The Projects

There are many background concerns, like networking the mix of UNIX workstations, IBM PCs and Apple Macintoshes that showed up at the Lab, or our constant study of the World Wide Web and its continuing developments.

But the center of the work is the deep and visionary ambition shared by both IntelligentPad and Xanadu, both built around the hope of empowerment and a better life for everyone. Today's Internet provides an astonishing new environment that can help bring these dreams to life.

The People

While the Lab members represent their individual companies with considerable devotion, as a team they share a great camaraderie and enthusiasm for Lab projects.

A remarkable aspect of the Lab is the high level of technical skill, and particular type of cleverness, that each team member brings to the project. We come from a variety of backgrounds: chemistry, precision engineering, mathematics, philosophy and film-making; but working together unifies us all.

Professor Tanaka's gentle guidance combines both traditional Japanese leadership qualities and administrative skills with an American-style imagination. He brings out the best in everybody. Ted Nelson, on the other hand, combines his American rush of kinetic and far-seeing inspirations, with a Japanese perfectionism in his writing and a Japanese level of caring about what he is working on.

One Suggestion

Perhaps the one most significant improvement every team member could make would be to learn more of the other's language, thus allowing for even deeper and more enriching communications

The Projects

There are many background concerns, like networking the mix of UNIX workstations, IBM PCs and Apple Macintoshes that showed up at the Lab, or our constant study of the World Wide Web and its continuing developments.

But the center of the work is the deep and visionary ambition shared by both IntelligentPad and Xanadu, both built around the hope of empowerment and a better life for everyone. Today's Internet provides an astonishing new environment that can help bring these dreams to life.

The People

While the Lab members represent their individual companies with considerable devotion, as a team they share a great camaraderie and enthusiasm for Lab projects.

A remarkable aspect of the Lab is the high level of technical skill, and particular type of cleverness, that each team member brings to the project. We come from a variety of backgrounds: chemistry, precision engineering, mathematics, philosophy and film-making: but working together unifies us all.

Professor Tanaka's gentle guidance combines both traditional Japanese leadership qualities and administrative skills with an American-style imagination. He brings out the best in everybody. Ted Nelson, on the other hand, combines his American rush of kinetic and far-seeing inspirations, with a Japanese perfectionism in his writing and a Japanese level of caring about what he is working on.

One Suggestion

Perhaps the one most significant improvement every team member could make would be to learn more of the other's language, thus allowing for even deeper and more enriching communications

IntelligentPad Strategic Issues

Ted Nelson, Research Fellow, Sapporo HyperLab

OBJECTIVES OF THIS PAPER

How can IntelligentPad best become a powerful, effective and widespread tool in tomorrow's computer world?

We need to understand the opportunities, possibilities and pressures of the present situation.

THE CURRENT SITUATION AND PROBLEM OF INTELLIGENTPAD

While IntelligentPad has some remarkable characteristics, it also has several tactical disadvantages and difficulties.

It also has unique powers.

Its competitive position is poor and getting worse.

IntelligentPad Strategic Issues

Ted Nelson, Research Fellow, Sapporo HyperLab

OBJECTIVES OF THIS PAPER

How can IntelligentPad best become a powerful, effective and widespread tool in tomorrow's computer world?

We need to understand the opportunities, possibilities and pressures of the present situation.

THE CURRENT SITUATION AND PROBLEM OF INTELLIGENTPAD

While IntelligentPad has some remarkable characteristics, it also has several tactical disadvantages and difficulties.

It also has unique powers.

Its competitive position is poor and getting worse.

UNDERSTANDING INTELLIGENTPAD

It is still hard to explain IntelligentPad. This is because there is nothing else like it in depth. It is all too easy to compare it superficially with--

WIDGET-BASED SYSTEMS

NextStep, and other visual construction systems, permit easy prototyping by the combination of screen pieces, or "widgets." This is what IntelligentPad may look like to a lot of people.

CONTEMPORARY EASY LANGUAGES

Visual Basic is the front-running easy language (note that it has nothing to do with the original "Basic" language). Other important easy computer languages are NextStep, HyperCard and SuperCard. Yet another system, which may become very important, is "GainMomentum" from Sybase.

A MUCH GREATER DREAM

IntelligentPad represents a much greater dream, as outlined by Profesor Tanaka in various writings and his videotape. This is the dream of "meme media," a system of what he calls "media-based" software construction that can become a world-wide system of commerce, with pads whose parts can be sold, taken apart, and re-used in whole or part.

TIME FOR THE LAUNCH

Everyone involved with IntelligentPad admires this dream and wants to help bring it about. However, concrete actions must be taken to bring it out into the world, and the details and suitability of these actions will determine its success.

Current Position: LIKE UNIX IN 1970, LIKE THE MACINTOSH IN 1984

IntelligentPad is the kernel, the starting point, of a potential new industry and system of software.

However, this has a number of historical precedents. I think it is most like Unix in 1970 and the Macintosh in 1984-- the first part, but only the first part, of what could be an extensive movement.

The success of Unix was based on its acceptance by many thousands of university students and faculty. This happened over a considerable period of time, and happened because Unix was offered free to educational institutions.

The success of the Macintosh was much less certain, and there have been several critical periods where it could have died: right after launch, in 1984;

UNDERSTANDING INTELLIGENTPAD

It is still hard to explain IntelligentPad. This is because there is nothing else like it in depth. It is all too easy to compare it superficially with--

WIDGET-BASED SYSTEMS

NextStep, and other visual construction systems, permit easy prototyping by the combination of screen pieces, or "widgets." This is what IntelligentPad may look like to a lot of people.

CONTEMPORARY EASY LANGUAGES

Visual Basic is the front-running easy language (note that it has nothing to do with the original "Basic" language). Other important easy computer languages are NextStep, HyperCard and SuperCard. Yet another system, which may become very important, is "GainMomentum" from Sybase.

A MUCH GREATER DREAM

IntelligentPad represents a much greater dream, as outlined by Professor Tanaka in various writings and his videotape. This is the dream of "meme media," a system of what he calls "media-based" software construction that can become a world-wide system of commerce, with pads whose parts can be sold, taken apart, and re-used in whole or part.

TIME FOR THE LAUNCH

Everyone involved with IntelligentPad admires this dream and wants to help bring it about. However, concrete actions must be taken to bring it out into the world, and the details and suitability of these actions will determine its success.

Current Position:
LIKE UNIX IN 1970,
LIKE THE MACINTOSH IN 1984

IntelligentPad is the kernel, the starting point, of a potential new industry and system of software.

However, this has a number of historical precedents. I think it is most like Unix in 1970 and the Macintosh in 1984-- the first part, but only the first part, of what could be an extensive movement.

The success of Unix was based on its acceptance by many thousands of university students and faculty. This happened over a considerable period of time, and happened because Unix was offered free to educational institutions.

The success of the Macintosh was much less certain, and there have been several critical periods where it could have died: right after launch, in 1984;

1987-1988; and about 1992. It was saved by precise tactical actions--

1984: the inclusion of two basic programs, MacPaint and MacWrite, in the original Macintosh package. These allowed beginners to do word processing and very unusual graphic art projects, and were a sign to users and developers of the machine's potential.

Furthermore, Apple's "Evangelist" and developer programs encouraged a number of firms to enter the Macintosh software market.

1987-8: the appearance of HyperCard and Adobe PhotoShop. These defined a new genre (hypermedia) and a new industry (graphic arts image processing and prepress), locking in two new niche markets.

1992: Apple decides to open the system to other manufacturers, assuring a survivable market.

This is my personal analysis, and may not be correct, but the main point is that specific steps were taken carefully to deal with the Macintosh's strategic and competitive situation.

In the same way, we must consider appropriate steps to advance the strategic and competitive situation of IntelligentPad.

These steps will take resources and attention.

WHAT IS MISSING

A number of key things are missing in the present situation of IntelligentPad--

EXPLANATION AND DOCUMENTATION

I have not yet seen any documentation except of a very obscure sort. I have seen no good writeup for beginners; I have seen no good writeup for experienced computer people, except the guideline to C++ classes, which tells almost nothing about IntelligentPad or its use.

MARKETING

I know of no broad-based analysis of the marketing problems of IntelligentPad.

BASIC APPLICATIONS ("frameworks")

As far as I know, there presently exists no set of programs made with IntelligentPad that an ordinary user can use for any practical purpose whatever.

1987-1988; and about 1992. It was saved by precise tactical actions--

1984: the inclusion of two basic programs, MacPaint and MacWrite, in the original Macintosh package. These allowed beginners to do word processing and very unusual graphic art projects, and were a sign to users and developers of the machine's potential.

Furthermore, Apple's "Evangelist" and developer programs encouraged a number of firms to enter the Macintosh software market.

1987-8: the appearance of HyperCard and Adobe PhotoShop. These defined a new genre (hypermedia) and a new industry (graphic arts image processing and prepress), locking in two new niche markets.

1992: Apple decides to open the system to other manufacturers, assuring a survivable market.

This is my personal analysis, and may not be correct, but the main point is that specific steps were taken carefully to deal with the Macintosh's strategic and competitive situation.

In the same way, we must consider appropriate steps to advance the strategic and competitive situation of IntelligentPad.

These steps will take resources and attention.

WHAT IS MISSING

A number of key things are missing in the present situation of IntelligentPad--

EXPLANATION AND DOCUMENTATION

I have not yet seen any documentation except of a very obscure sort. I have seen no good writeup for beginners; I have seen no good writeup for experienced computer people, except the guideline to C++ classes, which tells almost nothing about IntelligentPad or its use.

MARKETING

I know of no broad-based analysis of the marketing problems of IntelligentPad.

BASIC APPLICATIONS ("frameworks")

As far as I know, there presently exists no set of programs made with IntelligentPad that an ordinary user can use for any practical purpose whatever.

EXPLANATION & DOCUMENTATION

I have not yet seen any documentation that will give an ordinary computer person an understanding of IntelligentPad. (Such documentation is rumored to exist, and I hope this is the case.)

It is vital to get the system explained, and explained, and explained. Note that there is an enormous number of books available about popular software packages, such as WordPerfect, HyperCard or Unix (dozens or even hundreds of different titles). It is very important to have books written by third parties. I would say that a good rule of thumb is that there are "enough" books written about IntelligentPad when there is a book about it in the IPC "Dummies" series, i.e., *IntelligentPad for Dummies*. (Note that *Internet for Dummies*, by my friend John Levine, has sold about 150,000 copies in the last year.)

There is an immediate need for an official introduction, as well as the more advanced documentation underway.

Among other titles needed, for both articles and books:

IntelligentPad for Beginners

IntelligentPad for Basic Programmers

IntelligentPad for Database Clients

IntelligentPad on the Internet
IntelligentPad for Personal Use
Programming Primitive Pads in C++
Programming IntelligentPad Modifiers

These are just the beginning. We must get other authors and publishers working on the matter.

EXPLANATION & DOCUMENTATION

I have not yet seen any documentation that will give an ordinary computer person an understanding of IntelligentPad. (Such documentation is rumored to exist, and I hope this is the case.)

It is vital to get the system explained, and explained. and explained. Note that there is an enormous number of books available about popular software packages, such as WordPerfect, HyperCard or Unix (dozens or even hundreds of different titles). It is very important to have books written by third parties. I would say that a good rule of thumb is that there are "enough" books written about IntelligentPad when there is a book about it in the IPC "Dummies" series, i.e., IntelligentPad for Dummies. (Note that Internet for Dummies, by my friend John Levine, has sold about 150,000 copies in the last year.)

There is an immediate need for an official introduction, as well as the more advanced documentation underway.

Among other titles needed, for both articles and books:

IntelligentPad for Beginners

IntelligentPad for Basic Programmers

IntelligentPad for Database Clients

IntelligentPad on the Internet
IntelligentPad for Personal Use
Programming Primitive Pads in C++
Programming IntelligentPad Modifiers

These are just the beginning. We must get other authors and publishers working on the matter.

USABLE FRAMEWORKS

Definition of User Domains and Message Sets

There do not exist any complete, usable IntelligentPad environments for beginners or non-programmers.

It is vital to develop some kind of a beginner's world so that *someone* can start using IntelligentPad for *some* everyday activity.

Utilities

Utilities: programs that allow fundamental activities. These include printing (and especially PostScript and TrueType), Draw Utilities, conventional text formatting primitives, and much more.

APIs

and creative development of comp

APIs: Application Program Interfaces. These will be interfaces that allow other programs to tie in with IntelligentPad, either for input or output. They have yet to be defined.

MARKETING

Even if some versions of IntelligentPad are free, they still must be marketed. (Marketing applies to products at all price levels, even Zero.)

Marketing will mean
the definition of markets
the targeting of products for specific markets
the design of rollout and campaigns

Possible Markets

sdfsdfdf

USABLE FRAMEWORKS

Definition of User Domains and Message Sets

There do not exist any complete, usable IntelligentPad environments for beginners or non-programmers.

It is vital to develop some kind of a beginner's world so that someone can start using IntelligentPad for some everyday activity.

Utilities

Utilities: programs that allow fundamental activities. These include printing (and especially PostScript and TrueType), Draw Utilities, conventional text formatting primitives, and much more.

APIs

and creative development of comp

APIs: Application Program Interfaces. These will be interfaces that allow other programs to tie in with IntelligentPad, either for input or output. They have yet to be defined.

MARKETING

Even if some versions of IntelligentPad are free, they still must be marketed. (Marketing applies to products at all price levels, even Zero.)

Marketing will mean
the definition of markets
the targeting of products for specific markets
the design of rollout and campaigns

Possible Markets

sdfsdfdf

FURTHER DESIGN OF FRAMEWORKS

uter m

User Domains

What are main definitions of user domain?

FURTHER CONVENTIONS

en to be a laboratory

So many things are undecided as to the future uses of IntelligentPad: major uses of every kind and how they will tie together.

While we are all deeply impressed by Professor Tanaka's vision, making these aspects more specific will be a vital part of what happens.

FURTHER DESIGN OF FRAMEWORKS

User Domains

What are main definitions of user domain?

FURTHER CONVENTIONS

on to be a laboratory

So many things are undecided as to the future uses of IntelligentPad: major uses of every kind and how they will tie together.

While we are all deeply impressed by Professor Tanaka's vision, making these aspects more specific will be a vital part of what happens.

Market Preparation

Steps must be taken to prepare markets for a product. These have to do with making the public aware-- public relations and magazine placement; with involving opinion leaders-- key user placement; with defining the market and fine-tuning the product (often done with focus groups).

This is the sort of thing that Dentsu would charge a million dollars for, and rightly so.

Market Seeding (Business)

Database
Form-based process flow
xxx

Business
Market Seeding (Non-Business)

A key aspect of getting people to use something is seeding the market. Unix was seeded into universities; Macintoshes were seeded into universities and the graphic arts community.

SCHOOLS (University and Graduate School)

IntelligentPad was created at a university, and as an unusual software construction set it has special appeal to students-- especially if they can use it for thesis, project or research work.

Defining frameworks

University students and professors could also have practical frameworks produced for their lives.

SCHOOLS (University Administration)

Student Registration and record-keeping

SCHOOLS (Primary Administration)

Scheduling
Lesson Plans
Student Registration and record-keeping

Market Preparation

Steps must be taken to prepare markets for a product. These have to do with making the public aware-- public relations and magazine placement; with involving opinion leaders-- key user placement; with defining the market and fine-tuning the product (often done with focus groups).

This is the sort of thing that Dentsu would charge a million dollars for, and rightly so.

Market Seeding (Business)

Database
Form-based process flow
xxx

Business
Market Seeding (Non-Business)

A key aspect of getting people to use something is seeding the market. Unix was seeded into universities; Macintoshes were seeded into universities and the graphic arts community.

SCHOOLS (University and Graduate School)

IntelligentPad was created at a university, and as an unusual software construction set it has special appeal to students-- especially if they can use it for thesis, project or research work.

Defining frameworks

University students and professors could also have practical frameworks produced for their lives.

SCHOOLS (University Administration)

Student Registration and record-keeping

SCHOOLS (Primary Administration)

Scheduling
Lesson Plans
Student Registration and record-keeping

SCHOOLS (Primary Teachers)

Primary-school teachers could have special frameworks.

SCHOOLS (Primary Students)

Primary-school pupils could use it for programming simple applications, the way Logo was used at MIT.

SCHOOLS (Secondary Teachers)

Secondary-school teachers could have special frameworks.

SCHOOLS (Secondary Pupils)

Secondary-school pupils could use it for programming simple applications, the way Logo was used at MIT.

SCHOOLS (Secondary Administration)

Scheduling
Lesson Plans
Student Registration and record-keeping

FAMILIES

Scheduling
Medical records
Records of possessions
Address lists

Pre-Teens

Scheduling

TEEN-AGERS

Games
Practical: dating, phones, sports
MANGA
MUSIC

dministration)

Scheduling

dministration)

Scheduling

SCHOOLS (Primary Teachers)
Primary-school teachers could have special frameworks.

SCHOOLS (Primary Students)
Primary-school pupils could use it for programming simple applications, the
way Logo was used at MIT.

SCHOOLS (Secondary Teachers)
Secondary-school teachers could have special frameworks.

SCHOOLS (Secondary Pupils)
Secondary-school pupils could use it for programming simple applications,
the way Logo was used at MIT.

SCHOOLS (Secondary Administration)
Scheduling
Lesson Plans
Student Registration and record-keeping

FAMILIES
Scheduling
Medical records
Records of possessions
Address lists

Pre-Teens
Scheduling

TEEN-AGERS
Games
Practical: dating, phones, sports
MANGA
MUSIC

dministration)
Scheduling

dministration)
Scheduling

Market Infrastructure

DEVELOPERS AND SYSTEMS INTEGRATORS

If IntelligentPad is to become a serious contender in the software market, it needs to have an active community of developers, who have an active channel of communication to IntelligentPad sources.

MARKETERS AND REMARKETERS

"Marketers" will be those firms who are marketing products based on Intelligent-Pad. Remarketers will be taking those products and selling them to end users, or repackaging them for other target customers.

BUNDLERS

"Bundlers" will be those businesses which take IntelligentPad and put it in a machine, or a product. They might not advertise that IntelligentPad is in use, just as software developers do not ordinarily tell people what language their programs are based on.

Some bundlers may just sell " IntelligentPad machines" (just as Fintronics USA sells PC clones set up to run Linux). In that case they would be telling customers that IntelligentPad was inside.

CONSULTANTS

the HyperLab is

Consultants are among the most vital possible

SERVICE BUREAUS

the HyperLab is the HyperLab is

flourishing due to the merging of the best qualities of each of its component parts.

It has brought together the spirit of teamwork and cooperation so characteristic of Japan and the individualistic and creative energy of America

Market Infrastructure

DEVELOPERS AND SYSTEMS INTEGRATORS

If IntelligentPad is to become a serious contender in the software market, it needs to have an active community of developers, who have an active channel of communication to IntelligentPad sources.

MARKETERS AND REMARKETERS

"Marketers" will be those firms who are marketing products based on Intelligent-Pad. Remarketers will be taking those products and selling them to end users, or repackaging them for other target customers.

BUNDLERS

"Bundlers" will be those businesses which take IntelligentPad and put it in a machine, or a product. They might not advertise that IntelligentPad is in use, just as software developers do not ordinarily tell people what language their programs are based on.

Some bundlers may just sell " IntelligentPad machines" (just as Pintronics USA sells PC clones set up to run Linux). In that case they would be telling customers that IntelligentPad was inside.

CONSULTANTS

the HyperLab is
Consultants are among the most vital possible

SERVICE BUREAUS

the HyperLab is the HyperLab is
flourishing due to the merging of the best qualities of each of its component parts.

It has brought together the spirit of teamwork and cooperation so characteristic of Japan and the individualistic and creative energy of America

KILLER APS

A Browser Widget

THE CHOICES AHEAD

I do not believe that IntelligentPad will sell itself, or have any substantial impact without proper marketing and demo apps.

There have to be choices and they have to be funded.

Demo Apps

Something has to be built.
Like the original MacPaint and MacWrite

KEY DEMO APPS & FRAMEWORKS
Business?
School?
Personal?
Internet?

Market Selection

There

Apps

There

Market Preparation

There has to be Market Prep

Rollout

There has to be a Rollout, and it mustn't be botched.

Follow-On

There need to be continuing programs--
DEVELOPER PROGRAM
USER HOT LINE
USER GROUPS

KILLER APPS

A Browser Widget

THE CHOICES AHEAD

I do not believe that IntelligentPad will sell itself, or have any substantial impact without proper marketing and demo apps.

There have to be choices and they have to be funded.

Demo Apps

Something has to be built.
Like the original MacPaint and MacWrite

KEY DEMO APPS & FRAMEWORKS
Business?
School?
Personal?
Internet?

Market Selection

There

Apps

There

Market Preparation

There has to be Market Prep

Rollout

There has to be a Rollout, and it mustn't be botched.

Follow-On

There need to be continuing programs--
DEVELOPER PROGRAM
USER HOT LINE
USER GROUPS

Conclusion

I have presented some strong views about what I think needs to be done. Like many others, I want to see Professor Tanaka achieve his dream. I believe it will take considerable resources and planning. These notes are intended to share my thoughts on the matter.

Conclusion

I have presented some strong views about what I think needs to be done. Like many others, I want to see Professor Tanaka achieve his dream. I believe it will take considerable resources and planning. These notes are intended to share my thoughts on the matter.

Xuf. 99 REF
As delivered to Fujitsu
(to Okamoto via Takebe)
95.5.18
pm 730

Xanadu* and IntelligentPad

Ted Nelson, Research Fellow, Sapporo HyperLab

Note: "Xanadu" and the Flaming-X symbols are U.S. registered trademarks of Project Xanadu, 3020 Bridgeway #295, Sausalito CA 94965, USA.

Project Xanadu

Project Xanadu has for several decades been an ongoing development effort for a particular kind of back-end data service and electronic publishing. The objective has been to facilitate the creation and sharing of application-independent hypermedia objects that can be reworked, shared, re-used, published, and sold.

A principal intention of Project Xanadu has been to avoid the difficulties of permission in publishing, through a specific copyright approach now called Transcopyright. The published document consists of a list of pointers to transcopyrighted material, so that each user buys each piece from its original publisher. Thus nothing is misquoted, nothing is out of context (since the original context is available), and each publisher is paid a small amount for each use. But each publisher must give persmission for re-use in arbitrary new contexts.

There have been very ambitious attempts at implementation, the most recent by Autodesk, Inc. in the United States.

Xanadu Idealism

Xanadu has always been a very idealistic enterprise, concerned for freedom of information for everyone, and for simplifying and eliminating copyright as a problem, while retaining its benefits. The work has been well known and has excited people in many parts of the world.

I am presently developing specifications for the generalized Xanadu functions so that anyone can implement these types of function, although the trademark "Xanadu" will only be licensed to some.

Xanadu* and IntelligentPad

Ted Nelson, Research Fellow, Sapporo Hypertab

Note: "Xanadu" and the Flowing-X symbols are U.S. registered trademarks of Project Xanadu, 3020 Hillgaway #295, Sausalito CA 94965, USA.

Project Xanadu

Project Xanadu has for several decades been an ongoing development effort for a particular kind of back-end data service and electronic publishing. The objective has been to facilitate the creation and sharing of application-independent hypermedia objects that can be reworked, shared, re-used, published, and sold.

A principal intention of Project Xanadu has been to avoid the difficulties of permission in publishing, through a specific copyright approach now called Transcopyright. The published document consists of a list of pointers to transcopyrighted material, so that each user buys each piece from its original publisher. Thus nothing is misquoted, nothing is out of context (since the original context is available), and each publisher is paid a small amount for each use. But each publisher must give permission for re-use in advance any new contexts.

There have been very ambitious attempts at implementation, the most recent by Autodesk, Inc. in the United States.

Xanadu Identities

Xanadu has always been a very idealistic enterprise, concerned for freedom of information for everyone, and for simplifying and eliminating copyright as a problem, while retaining its benefits. The work has been well known and has excited people in many parts of the world.

I am presently developing specifications for the generalized Xanadu functions so that anyone can implement these types of function, although the trademark "Xanadu" will only be licensed to some

Xanadu Functions

The Xanadu Project has always been two things; a kind of local service for data, keeping re-used materials clearly identified; and a kind of publishing service, also keeping re-used materials clearly identified.

A Xanadu server keeps track of contents, links and transclusions. In the Xanadu way of thinking, a link is a connection between materials which are different, and a transclusion is a connection between things which are the same (i.e., a virtual copy having the same identity).

This helps in the intercomparison of documents, versions and re-uses of all kinds.

Xanadu Back-End Data Functions

Locally, Xanadu function is defined as a specific data service around which applications can be built for linked material (hypermedia) and transcluded (or re-used) material. A Xanadu server keeps track of contents and specific structural relations (links and transclusions), despite evolving changes as the material is edited.

Xanadu data consists of primary elements and objects created by users-- text, pictures, video, paintfiles, etc. These may be composited with various structural relationships, and by transclusion may be re-used in virtual objects of all kinds.

The Xanadu Front-End Client

In the Xanadu model, the front end is a computer which sends for and assembles portions of contents, presents them to the user, and interacts with the data.

The Xanadu Publishing System

Anyone may open a publishing node, and anyone may publish. Publishing involves a long-term commitment to keep the material available, and permission for its republication under the transcopyright agreement.

If the user chooses to follow a hyperlink on the screen, his front-end system checks the price of the materials. If the price is acceptable according to a pre-programmed range, the purchase transaction is carried out automatically without the user's noticing it. The portion which is delivered from the publisher is now owned by the user. The user has the option of retaining the piece, which he now owns, as part of his private library.

Furthermore, any user may republish anything virtually through the transcopyright method.

Xanadu Functions

The Xanadu Project has always been two things; a kind of local service for data, keeping re-used materials clearly identified; and a kind of publishing service, also keeping re-used materials clearly identified.

A Xanadu server keeps track of contents, links and transclusions. In the Xanadu way of thinking, a link is a connection between materials which are different, and a transclusion is a connection between things which are the same (i.e., a virtual copy having the same identity).

This helps in the intercomparison of documents, versions and re-uses of all kinds.

Xanadu Back-End Data Functions

Locally, Xanadu function is defined as a specific data service around which applications can be built for linked material (hypermedia) and transcluded for re-used material. A Xanadu server keeps track of contents and specific structural relations (links and transclusions), despite evolving changes as the material is edited.

Xanadu data consists of primary elements and objects created by users-- text pictures, video, painfile, etc. These may be composited with various structural relationships; and by transclusion may be re-used in virtual objects of all kinds.

The Xanadu Front-End Client

In the Xanadu model, the front end is a computer which sends for and assembles portions of contents, presents them to the user, and interacts with the data.

The Xanadu Publishing System

Anyone may open a publishing node, and anyone may publish. Publishing involves a long-term commitment to keep the material available, and permission for its republication under the transcopyright agreement.

If the user chooses to follow a hyperlink on the screen, his front-end system checks the price of the materials. If the price is acceptable according to a pre-programmed range, the purchase transaction is carried out automatically without the user's noticing it. The portion which is delivered from the publisher is now owned by the user. The user has the option of resulting the piece, which he now owns, as part of his private library.

Furthermore, any user may republish anything virtually through the transcopyright method.

IntelligentPad and its Ideals

IntelligentPad is a highly-capable interactive system toolkit with remarkable properties. It allows the creation of both back-end and client systems, and can function seamlessly across a network, including the Internet.

Professor Tanaka, the developer of IntelligentPad, has a very similar idealism to mine with regard to the distribution and availability of information, and so he has designed IntelligentPad with a model of distribution and access that is similar to Xanadu.

In principle, then, it should be possible to use IntelligentPad for Xanadu-class functions, to construct both back-end servers and front-end clients. Either of these applications would require a serious effort, but both are possible.

Used as the basis for a front-end application, an IntelligentPad client could function on the Internet, for editing, compositing, publishing and republishing materials; and for viewing, interacting with, and purchasing materials from other publishers across the Internet, and storing them in the user's local system.

Such a front end could use either IntelligentPad's native paste-and-peel interface, or could be adapted to other interfaces.

Used as the basis of a Xanadu-class back-end server, IntelligentPad could be used to program both a server, and a delivery medium, absorbing various network functions.

IntelligentPad and Xanadu on Today's Internet

Today's Internet presents many new opportunities for tomorrow's software, and a highly competitive environment for world-wide commerce. I believe IntelligentPad can become a significant software force in that environment. I believe that its combination with Xanadu may create a powerful synergistic effect.

Clean Data (without special cases or embedded codes)

constructed of pure data elements, links and transclusions

This has the special advantage IntelligentPad's built-in paste-and-peel metaphor

in addition, the relation of transclusion, or virtual embedding of instances in new contexts which can be compared with the original

Various structural relations

Basic service for hypermedia: links and data

Principal intentions of the Xanadu design have included the clear intercomparison of different uses of materials, and to avoid the difficulties of permission in publishing through a specific copyright approach.

Various implementations. Most recent, backed by Autodesk, now licensed to Filoli, Inc.

World Wide Web took part of the design, but simplified it

Clean Data (without special cases or embedded codes)

constructed of pure data elements, links and transclusions

This has the special advantage IntelligentPad's built-in paste-and-peel metaphor

In addition, the relation of transclusion, or virtual embedding of instances in new contexts which can be compared with the original

Various structural relations

Basic service for hypermedia: links and data

Various implementations . Most recent . backed by Autodesk, now licensed to Filoli, Inc.

World Wide Web took part of the design, but simplified it

Principal intentions of the Xanadu design have included the clear interconnection of different uses of materials, and to avoid the difficulties of permission in publishing through a specific copyright approach.

[turned in for Lab Open House title]

tpl.title.7.13

TRANSPARALLEL MEDIA AND TOMORROW'S LITERATURE

Theodor Holm Nelson
Project Xanadu®
and
Sapporo HyperLab
e-mail ted@xanadu.net

TRANSPARALLEL MEDIA AND TOMORROW'S LITERATURE

REF

MANUSCRIPTS USED
FOR PRESENTATION
TO IPC STEERING COMMITTEE,
94.5.25.

---

2 pages of sticky notes, Ø Ø & Ø

Incomplete draft 6 of
"IntelligentPad Strategic Issues"
(through page 5)

Previous draft (noodle form),
draft 5 → 6.

MANUSCRIPTS USED
FOR PRESENTATION
TO 1964 STEERING COMMITTEE,
pp. 1-25.

---

2 pages of title, notes, table of
Incomplete draft B of
"Intelligent Strategic Issues"
(Theory) (page 5)
Previous draft (needle form),
draft 1-4.

UPBEAT!

SUG'd
$ pass of
dead ideas

FEAR
THIS GROUP
MAY BE
TOO
OPTIMISTIC

DIDN'T KNOW THESE PEOPLE WANT TO HELP

IMPRESSIONS — tm perspective of CAVS

&

MY EXPERIENCE IN SOFTWARE

HELP WITH

MARKETING & ROLLOUT

YOU ONLY GET ONE ROLLOUT

MAXIMIZE IMPACT
Make sure it's real

Expert in Everyone
KOFIC
GEN. MAGIC
ORACLE
SYBASE

IF YOU COMPETE WITH EVERYONE PICK YOUR TARGETS ONE BY ONE

FREE CD-ROM?

SHAREWARE AT EGGHEAD

XANADU APPS

It's PICTURE OF COMMERCE.

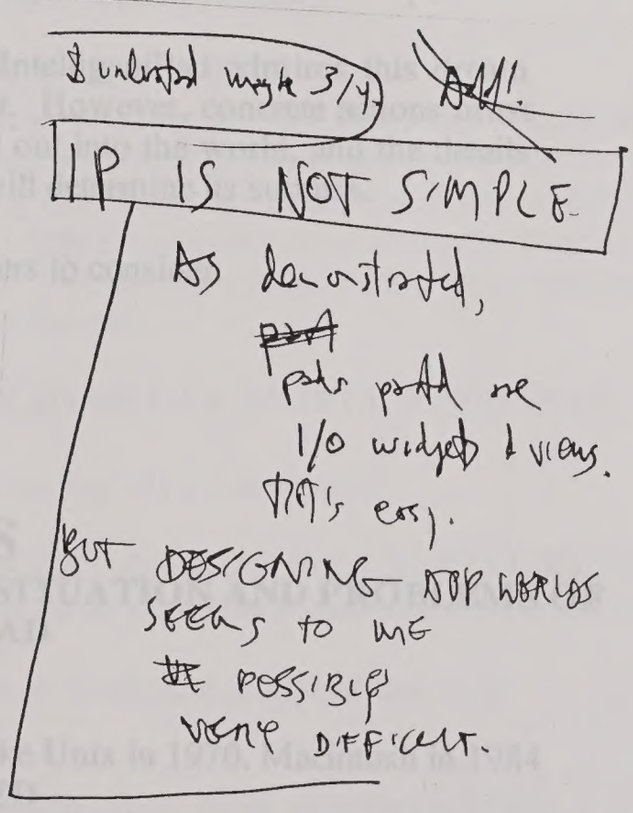
MM APPS
CD-ROM builder
(Director de capitol)

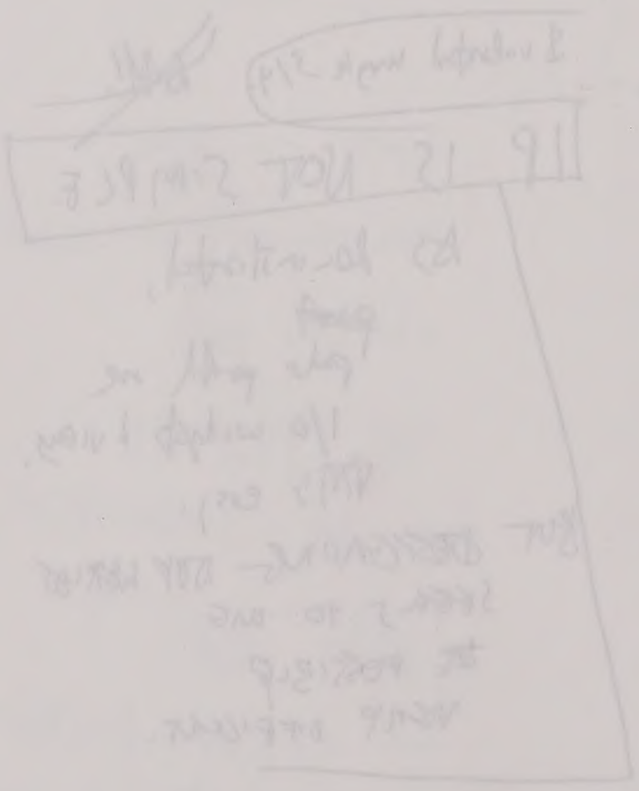
MACROMIND D

All.
Remarks

NO FINAL VERSION
OF SOFTWARE.

A LONG WAY
ON IT'S INTUITION





IntelligentPad Strategic Issues

Ted Nelson, Research Fellow, Sapporo HyperLab

OBJECTIVES OF THIS PAPER

How can IntelligentPad best become a powerful, effective and widespread tool in tomorrow's computer world?

We need to understand the opportunities, possibilities and pressures of the present situation.

IntelligentPad represents a wonderful vision, as outlined by Profesor Tanaka in various writings and his videotape. This is the dream of "meme media," a system of what he calls "media-based" software construction that can become a world-wide system of commerce, with pads whose parts can taken apart, and re-used in whole or part, and sold in whole or part by anyone, creating a vast system of commerce.

Everyone involved with IntelligentPad admires this dream and wants to help bring it about. However, concrete actions must be taken to bring IntelligentPad out into the world, and the details and suitability of these actions will determine its success.

This paper lists some matters to consider.

CONTENTS

THE CURRENT SITUATION AND PROBLEMS OF INTELLIGENTPAD
- What it is
- Competition
- Situation now: like Unix in 1970, Macintosh in 1984

WHAT IS NEEDED
- Explanation and Documentation
- Basic Applications
- Marketing
- Distribution Channels; Market Preparation

THE CHOICES AHEAD

SUMMARY AND CONCLUSION

IntelligentPad
Strategic Issues

THE CURRENT SITUATION AND PROBLEMS OF INTELLIGENTPAD

IntelligentPad has unique powers and remarkable characteristics.

It also has certain tactical disadvantages and difficulties, and it is in a difficult competitive position.

WHAT IS IT EXACTLY?

The strategic situation depends on what you think IntelligentPad actually IS. IntelligentPad is unique; but it rsembles various other products in various ways.

WIDGET-BASED SYSTEMS AND SOFTWARE CONSTRUCTION SETS

NextStep, and other visual construction systems, permit easy prototyping by the combination of screen pieces, or "widgets." This is what IntelligentPad may look like to a lot of people.

As a construction set, IntelligentPad seems to consist of--

INPUT DATA AND WIDGETS
OUTPUT DISPLAYS AND WIDGETS;
A PIPING METHOD.

Unfortunately, most such systems have failed on the market. An exception seems to be NextStep, which has a high-end niche market.

INTELLIGENTPAD AS A SIMPLE LANGUAGE

IntelligentPad gives the impression of being an easy computer language, but it has not yet been developed in that direction. There is plenty of competition there.

Visual Basic is the front-running easy language (note that it has nothing to do with the original "Basic" language). Other important easy computer languages are NextStep, HyperCard and SuperCard. Yet another system, which may become very important, is "GainMomentum" from Sybase.

INTELLIGENTPAD AS A BACKBONE FOR SIMPLE PERSONAL SYSTEMS

IntelligentPad gives that impression but not yet developed.

INTELLIGENTPAD AS A FRAMEWORK FOR INDUSTRIAL SYSTEMS

We believe that IntelligentPad can be an excellent framework for Script-based or document-based factory flow models.

This application has been quite well developed in at least one case.

GREATEST STRENGTH: INTELLIGENTPAD AS A FRAMEWORK FOR DISTRIBUTED SYSTEMS

This where IntelligentPad shines.

IntelligentPad is already able to support distributed applications across the Internet, whether in one office or locations thousands of miles apart.

THE CURRENT SITUATION AND PROBLEMS OF INTELLIGENTPAD

The system can support multiple stations with coordinated graphical activities, and the network transmission of both data and pads between distributed porrtions of applications.

POSSIBLE GIVEAWAYS ON THE WEB

We have been putting resources into a Mosaic-like browser, but it is highly unlikely that an IP-based browser will replace Mosaic or Netscape. Imitating them means being secondary; offering different features may mean an achievable niche market.

A number of possibilities come to mind. Two are:

A "Web Companion" program that will save URLS and your comments on them, even file Web pages, and convert the pictures where desired.

A simple graphical exchange system, separate from e-mail, that would handle simple animations.

GREAT COMPETITION

There is great competition throughout the software field. Even if a product is free, or the best, it must still compete.

1. Competition against IntelligentPad in the Software Market

The problems of the software market are becoming known and understood, at least the way the market works now. Many of us would like it to work differently, but that is a separate issue.

Market share, and brand name recognition, are fiercely captured and defended.

Tens of thousands of software developers give out their works through Shareware because they have no other way.

Competing with IntelligentPad are expensively-developed and expensively-marketed--

Computer Languages Languages, Construction Sets

Database Products

Personal Applications

In order to compete against these, an explicit strategy must be chosen and carried out.

2. Competition against IntelligentPad on the Internet

Everything is still free on the Internet. That does not mean it isn't competititve: it's competitive in a very unusual way, presently the competition is still to *give things away.*

GREAT COMPETITION

There is great competition throughout the software field. Even if a product is free, or the best, it must still compete.

1. Competition against IntelligentPad in the Software Market

The problems of the software market are becoming known and understood, at least the way the market works now. Many of us would like it to work differently, but that is a separate issue.

Market share, and brand name recognition, are fiercely captured and defended.

Tens of thousands of software developers give out their works through Shareware because they have no other way.

Competing with IntelligentPad are expensively-developed and expensively-marketed--

Computer Languages, Constructive Sets

Database Products

Personal Applications

In order to compete against these, an explicit strategy must be chosen and carried out.

2. Competition against IntelligentPad on the Internet

Everything is still free on the Internet. That does not mean it isn't competitive: it's competitive in a very unusual way, presently the competition is still to give things away.

Current Position:
LIKE UNIX IN 1970,
LIKE THE MACINTOSH IN 1984

IntelligentPad is the kernel, the starting point, of a potential new industry and system of software.

However, this has a number of historical precedents. I think it is most like Unix in 1970 and the Macintosh in 1984-- it is the beginning, but only the beginning, of what could be an extensive movement.

The success of Unix was based on its acceptance by many thousands of university students and faculty. This happened over a considerable period of time, and happened because Unix was offered free to educational institutions.

The success of the Macintosh was much less certain, and there have been several critical periods where it could have died: right after launch, in 1984; 1987-1988; and about 1992. It was saved by precise tactical actions--

1984: the inclusion of two basic programs, MacPaint and MacWrite, in the original Macintosh package. These allowed beginners to do word processing and very unusual graphic art projects, and were a sign to users and developers of the machine's potential.

Furthermore, Apple's "Evangelist" and developer programs encouraged a number of firms to enter the Macintosh software market.

1987-8: the appearance of HyperCard and Adobe PhotoShop. These defined a new genre (hypermedia) and a new industry (graphic arts image processing and prepress), locking in two new niche markets.

1992: Apple decides to open the system to other manufacturers, assuring a survivable market.

This is my personal analysis, and may not be correct, but the main point is that specific steps were taken carefully to deal with the Macintosh's strategic and competitive situation.

In the same way, we must consider appropriate steps to advance the strategic and competitive situation of IntelligentPad.

These steps will take resources and attention.

Current Position:
LIKE UNIX IN 1970,
LIKE THE MACINTOSH IN 1984

IntelligentPad is the kernel, the starting point, of a potential new industry and system of software.

However, this has a number of historical precedents. I think it is most like Unix in 1970 and the Macintosh in 1984-- it is the beginning, but only the beginning, of what could be an extensive movement.

The success of Unix was based on its acceptance by many thousands of university students and faculty. This happened over a considerable period of time, and happened because Unix was offered free to educational institutions.

The success of the Macintosh was much less certain, and there have been several critical periods where it could have died: right after launch, in 1984; 1987-1988; and about 1992. It was saved by precise tactical actions--

1984: the inclusion of two basic programs, MacPaint and MacWrite, in the original Macintosh package. These allowed beginners to do word-processing and very unusual graphic art projects, and were a sign to users and developers of the machine's potential.

Furthermore, Apple's "Evangelist" and developer programs encouraged a number of firms to enter the Macintosh software market:

1987-8: the appearance of HyperCard and Adobe PhotoShop. These defined a new genre (hypermedia) and a new industry (graphic arts image processing and prepress), locking in two new niche markets.

1992: Apple decides to open the system to other manufacturers, assuring a survivable market.

This is my personal analysis, and may not be correct, but the main point is that specific steps were taken carefully to deal with the Macintosh's strategic and competitive situation.

In the same way, we must consider appropriate steps to advance the strategic and competitive situation of IntelligentPad.

These steps will take resources and attention.

IntelligentPad Strategic Issues

Ted Nelson, Research Fellow, Sapporo HyperLab

OBJECTIVES OF THIS PAPER

How can IntelligentPad best become a powerful, effective and widespread tool in tomorrow's computer world?

We need to understand the opportunities, possibilities and pressures of the present situation.

IntelligentPad represents a wonderful vision, as outlined by Profesor Tanaka in various writings and his videotape. This is the dream of "meme media," a system of what he calls "media-based" software construction that can become a world-wide system of commerce, with pads whose parts can be sold, taken apart, and re-used in whole or part,

Everyone involved with IntelligentPad admires this dream and wants to help bring it about. However, concrete actions must be taken to bring IntelligentPad out into the world, and the details and suitability of these actions will determine its success.

CONTENTS

THE CURRENT SITUATION AND PROBLEMS OF INTELLIGENTPAD
- What it is
- Competition
- Situation now: like Unix in 1970, Macintosh in 1984

WHAT IS NEEDED
- Explanation and Documentation
- Basic Applications
- Marketing
- Distribution Channels; Market Preparation

THE CHOICES AHEAD

SUMMARY AND CONCLUSION

IntelligentPad Strategic Issues

THE CURRENT SITUATION AND PROBLEMS OF INTELLIGENTPAD

IntelligentPad has unique powers and remarkable characteristics.

It also has several tactical disadvantages and difficulties, and it is in a difficult competitive position.

The Competitive Situation

Depends on what you think IntelligentPad IS

WHAT IS IT EXACTLY?

The competitive situation depends on what you think IntelligentPad actually IS.

The strategic situation depends on what you think IntelligentPad actually IS. It is unique; but it resembles some other prod's in some ways

~~IntelligentPad is superficially similar to certain products on the market~~

WIDGET-BASED SYSTEMS AND CONSTRUCTION SETS

As a Construction Set — ≤ HP

INPUT ~~AND OUPUT~~
DATA AND WIDGETS,

OUTPUT DISPLAYS
AND WIDGETS

PIPING METHOD

APR 2 5 1995

So far # not very successful.

NextStep, and other visual construction systems, permit easy prototyping by the combination of screen pieces, or "widgets." This is what IntelligentPad may look like to a lot of people.

APR 2 5 1995

CONTEMPORARY EASY LANGUAGES

Visual Basic is the front-running easy language (note that it has nothing to do with the original "Basic" language). Other important easy computer languages are NextStep, HyperCard and SuperCard. Yet another system, which may become very important, is "GainMomentum" from Sybase.

INTELLIGENTPAD AS A BACKBONE FOR SIMPLE PERSONAL SYSTEMS

Gives that impression but not yet developed

INTELLIGENTPAD AS A FRAMEWORK FOR INDUSTRIAL SYSTEMS

Script-based or document-based factory flow models

MERGET STRENGTH: NET

INTELLIGENTPAD AS A FRAMEWORK FOR DISTRIBUTED SYSTEMS

~~Fusion~~ network fusion, or data between distributed portions & ppl.

***This where it shines.
Large-scale.

Freebie software frobs —

Possible but not

Software browser
Hyllyn wide ly do replace Mosaic or Netscape
Niche market ~~for~~ OFFERING INTERNET [illegible]

WWW URL [illegible], [illegible], & f.b & Thumbnails.

1. Competition with IntelligentPad in the Software Market

The problems of the software market are becoming known and understood, at least the way the market works now. Many of us would like it to work differently, but that is a separate issue.

Tens of thousands of software developers give out their works through Shareware because they have no other way

Competitive Languages

Competitive Database Products

Competitive Personal Applications

2. Competition with IntelligentPad on the Internet

Everything is still free on the Internet. That does not mean it isn't competititve: it's competitive in a very unusual way, competition to give things away.

Current Position:
LIKE UNIX IN 1970,
LIKE THE MACINTOSH IN 1984

IntelligentPad is the kernel, the starting point, of a potential new industry and system of software.

However, this has a number of historical precedents. I think it is most like Unix in 1970 and the Macintosh in 1984-- the first part, but only the first part, of what could be an extensive movement.

The success of Unix was based on its acceptance by many thousands of university students and faculty. This happened over a considerable period of time, and happened because Unix was offered free to educational institutions.

The success of the Macintosh was much less certain, and there have been several critical periods where it could have died: right after launch, in 1984; 1987-1988; and about 1992. It was saved by precise tactical actions--

1984: the inclusion of two basic programs, MacPaint and MacWrite, in the original Macintosh package. These allowed beginners to do word processing and very unusual graphic art projects, and were a sign to users and developers of the machine's potential.

Furthermore, Apple's "Evangelist" and developer programs encouraged a number of firms to enter the Macintosh software market.

1987-8: the appearance of HyperCard and Adobe PhotoShop. These defined a new genre (hypermedia) and a new industry (graphic arts image processing and prepress), locking in two new niche markets.

1992: Apple decides to open the system to other manufacturers, assuring a survivable market.

This is my personal analysis, and may not be correct, but the main point is that specific steps were taken carefully to deal with the Macintosh's strategic and competitive situation.

In the same way, we must consider appropriate steps to advance the strategic and competitive situation of IntelligentPad.

These steps will take resources and attention.

Current Position:
LIKE UNIX IN 1970,
LIKE THE MACINTOSH IN 1984

IntelligentPad is the kernel, the starting point, of a potential new

industry and system of software.

However, this has a number of historical precedents. I think it is
most like Unix in 1970 and the Macintosh in 1984--
the first part, of what could be an extensive movement.

The success of Unix was based on its acceptance by many thousand
of university students and faculty. This happened over a considerable
period of time, and happened because Unix was offered free to educational
institutions.

The success of the Macintosh was much less certain, and there have
been several critical periods where it could have died: right after launch, in
1984; 1987-1988, and about 1992. It was saved by precise tactical
actions--

1984: the inclusion of two basic programs, MacPaint and
MacWrite, in the original Macintosh package. These allowed
beginners to do word processing and very unusual graphic art
projects, and were a sign to users and developers of the
machine's potential.

Furthermore, Apple's "Evangelist" and developer
programs encouraged a number of firms to enter the Macintosh
software market.

1987-8: the appearance of HyperCard and Adobe Photoshop.
These defined a new genre (hypermedia) and a new industry
(graphic arts image processing and prepress), locking in two
new niche markets.

1992: Apple decides to open the system to other manufacturers,
assuring a survivable market.

This is my personal analysis, and may not be correct, but the main
point is that specific steps were taken carefully to deal with the
Macintosh's strategic and competitive situation.

In the same way, we must consider appropriate steps to advance the
strategic and competitive situation of IntelligentPad.

These steps will take resources and attention.

WHAT IS MISSING

A number of key things are missing in the present situation of IntelligentPad--

EXPLANATION AND DOCUMENTATION

I have not yet seen any documentation except of a very obscure sort. I have seen no good writeup for beginners; I have seen no good writeup for experienced computer people, except the guideline to C++ classes, which tells almost nothing about IntelligentPad or its use.

BASIC APPLICATIONS ("frameworks")

As far as I know, there presently exists no set of programs made with IntelligentPad that an ordinary user can use for any practical purpose whatever.

MARKETING

I know of no broad-based analysis of the marketing problems of IntelligentPad.

DISTRIBUTION CHANNELS (Marketing Infranstructure)

While steps are being taken to put in place a particular method of component marketing, there is as yet no plan for developing organizational channels and contacts with people in existing industries who will uses these channels, and there has not as yet been an effort to define who IntelligentPad could use other channels which already exist.

I will discuss each of these in turn.

---

1. EXPLANATION & DOCUMENTATION

I have not yet seen any documentation that will give an ordinary computer person an understanding of IntelligentPad. (Such documentation is rumored to exist, and I hope this is the case.)

It is vital to get the system explained, and explained, and

8

WHAT IS MISSING

A number of key things are missing in the present situation of IntelligentPad--:

EXPLANATION AND DOCUMENTATION

I have not yet seen any documentation except of a very obscure sort. I have seen no good writeup for beginners; I have seen no good writeup for experienced computer people, except the underlining C++ classes, which tells almost nothing about IntelligentPad or its use.

BASIC APPLICATIONS ("Frameworks")

As far as I know, there presently exists no set of programs made with IntelligentPad that an ordinary person can use for any practical purpose whatever.

MARKETING

I know of no broad-based analysis of the marketing problems of IntelligentPad.

DISTRIBUTION CHANNELS (Marketing Infrastructure)

Steps are being taken to put in place a Darwinian method of component marketing, but there is as yet no plan--let alone developing organizational channels and contacts with people in existing industries who will uses these channels, and there has not as yet been an effort to define who IntelligentPad could use other channels which already exist.

I will discuss each of these in turn.

EXPLANATION & DOCUMENTATION

I have not yet seen any documentation that will give an ordinary computer person an understanding of IntelligentPad. (Such documentation is rumored to exist, and I hope this is the case.)

It is vital to get the system explained, and explained, and

explained. But the documentation should not merely come from the developers of the project. It is very important to have books written by third parties.

The more documentation from other publishers, the better for the product; that is how the market works today. There is an enormous number of books available about popular software packages, such as WordPerfect, HyperCard or Unix. Dozens or even hundreds of different titles.

(Note that *Internet for Dummies*, by my friend John Levine, has sold about 150,000 copies in the last year, but *Unix for Dummies*, by John and his sister, sold less than 50,000.)

There is an immediate need for an official introduction, as well as the more advanced documentation underway.

But articles and books on all of the following will be needed.

IntelligentPad for Beginners
IntelligentPad for Basic Programmers
IntelligentPad for Database Clients
IntelligentPad on the Internet
IntelligentPad for Personal Use
Programming Primitive Pads in C++
Programming IntelligentPad Modifiers

These are just the beginning. So the problem is how to get other authors and publishers working on the matter.

I think we can say there are "enough" books written about IntelligentPad when there is a book about it in the IPC *Dummies* series, i.e., *IntelligentPad for Dummies*. There are dozens of books in the Dummies series, about every kind of popular software (for example, *Microsoft Word for Dummies*, *Unix for Dummies*). Therefore the existence of a "Dummies" book means that a sofware package has become widely accepted and is considered stable and legitimate.

explained. But the documentation should not merely come from the developers of the project. It is very important to have books written by third parties.

The more documentation from other publishers, the better for the product; that is how the market works today. There is an enormous number of books available about popular software packages, such as WordPerfect, HyperCard or Unix dozens or even hundreds of different titles.

(Note that *Network for Dummies*, by my friend John Levine, has sold about 150,000 copies in the last year, but *Unix for Dummies*, by John and his sister, sold less than 50,000.)

There is an immediate need for an official introduction, as well as the more advanced documentation underway.

Our articles and books on all of the following will be needed:

IntelligentPad for Beginners

IntelligentPad for Basic Programmers

IntelligentPad for Database Clients

IntelligentPad on the Internet
IntelligentPad for Personal Use
Programming Primitive Pads in C++
Programming IntelligentPad Modifiers

These are just the beginning. So the problem is how to get other authors and publishers working on the matter.

I think we can say there are "enough" books written about IntelligentPad when there is a book about it in the IDG *Dummies* series, i.e., *IntelligentPad for Dummies*. There are dozens of books in the Dummies series, about every kind of popular software (for example, *Microsoft Word for Dummies*, *Unix for Dummies*). Therefore the existence of a "Dummies" book means that a software package has become widely accepted and is considered stable and legitimate.

USABLE FRAMEWORKS

Definition of User Domains and Message Sets

There do not exist any complete, usable IntelligentPad environments for beginners or non-programmers.

It is vital to develop some kind of a beginner's world so that *someone* can start using IntelligentPad for *some* everyday activity.

User Domains

~~What are main definitions of user domain?~~

FURTHER CONVENTIONS

MARKETING

Even if some versions of IntelligentPad are free, they still must be marketed. (Marketing applies to products at all price levels, even Zero.)

Marketing will mean :
the definition of markets
the targeting of products for specific markets
the design of rollout and campaigns

Possible Markets

There are many possible markets. They should all be considered before a decision is made on where to put the most effort.

SCHOOLS: University and Graduate School

IntelligentPad was created at a university, and as an unusual software construction set it has special appeal to students-- especially if they can use it for thesis, project or research work.

Defining frameworks

University students and professors could also have practical frameworks produced for their lives.

SCHOOLS: ~~University Administration~~

USABLE FRAMEWORKS

Definition of User Domains and Message Sets

There do not exist any complete, usable IntelligentPad environments for beginners or non-programmers.

It is vital to develop some kind of a beginner's world so that someone can start using IntelligentPad for some everyday activity.

User Domains

What are main definitions of user domain?

FURTHER CONVENTIONS

MARKETING

Even if some versions of IntelligentPad are free, they still must be marketed. (Marketing applies to products at all price levels, even zero.)

Marketing will mean :
the definition of markets
the targeting of products for specific markets
the design of rollout and campaigns

Possible Markets

There are many possible markets. They should all be considered before a decision is made on where to put the most effort.

SCHOOLS (University and Graduate School)

IntelligentPad was created in a university, and as an unusual software construction set it has special appeal to students -- especially if they can use it for thesis, project or research work.

Defining frameworks

University students and professors could also have practical frameworks produced for their lives.

SCHOOLS (University Administration)

Student Registration and record-keeping

SCHOOLS (Primary Administration)
Scheduling
Lesson Plans
Student Registration and record-keeping

SCHOOLS (Primary Teachers)
Primary-school teachers could have special frameworks.

SCHOOLS (Primary Students)
Primary-school pupils could use it for programming simple applications, the way Logo was used at MIT.

SCHOOLS (Secondary Teachers)
Secondary-school teachers could have special frameworks.

SCHOOLS (Secondary Pupils)
Secondary-school pupils could use it for programming simple applications, the way Logo was used at MIT.

SCHOOLS (Secondary Administration)
Scheduling
Lesson Plans
Student Registration and record-keeping

FAMILIES
Scheduling
Medical records
Records of possessions
Address lists
Video

Pre-Teens
Scheduling
Games

TEEN-AGERS
Games
Practical: dating, phones, sports
MANGA
MUSIC
GAME DESIGN
Audio
Video

WHAT INSIDE WANT TO DO

CD-ROM BUILDER

MSC. ADULTS
HM, MM BUILDING — IT'S WHAT ALL WANT TO DO

Market Infrastructure

THE IDEAL CHANNEL

It is part of the idea of IntelligentPad that it will be used for development and sale by independent programmers everywhere. This is quite possible. Every tool is used by inddependent programmers for development and sale, and IntelligentPad should also.

However, we should not neglect conventional channels which are already available.

DEVELOPERS AND SYSTEMS INTEGRATORS

If IntelligentPad is to become a serious contender in the software market, it needs to have an active community of developers, who have an active channel of communication to IntelligentPad sources.

PROGRAMMERS AND CONSULTANTS;

PROGRAMMERS AND consultants are among the most vital possible participants in IntelligentPad.

MARKETERS AND REMARKETERS;

"Marketers" will be those firms who are marketing products based on Intelligent-Pad. Remarketers will be taking those products and selling them to end users, or repackaging them for other target customers.

Some marketers may just sell " IntelligentPad machines" (just as Fintronics USA sells PC clones set up to run Linux). In that case they would be telling customers that IntelligentPad was inside.

BUNDLERS

"Bundlers" will be those businesses which take IntelligentPad and put it inside a machine, or a product. They might not advertise that IntelligentPad is in use, just as software developers do not ordinarily tell people what language their programs are based on.

SERVICE BUREAUS

Service bureaus, or companies which perform services using software of their own choice or creation, should be an important outlet for IntelligentPad applcations.

Market Preparation

Market Infrastructure

THE IDEAL CHANNEL

It is part of the idea of IntelligentPad that it will be used for development and sale by independent programmers everywhere. This is quite possible. Every tool is used by independent programmers for development and sale, and IntelligentPad should also.

However, we should not neglect conventional channels which are already available.

DEVELOPERS AND SYSTEMSINTEGRATORS

If IntelligentPad is to become a serious contender in the software market, it needs to have an active community of developers, who have an active channel of communication to IntelligentPad sources.

PROGRAMMERS AND CONSULTANTS

PROGRAMMERS AND consultants are among the most vital possible participants in IntelligentPad.

MARKETERS AND REMARKETERS; SOLUTIONS

"Marketers" will be those firms who are marketing products based on Intelligent-Pad. Remarketers will be taking those products and selling them to end users, or repackaging them for other target customers.

Some marketers may just sell " IntelligentPad machines" (just as Printronics USA sells PC clones set up to run Linux). In that case they would be telling customers that IntelligentPad was inside.

BUNDLERS

"Bundlers" will be those businesses which take IntelligentPad and sell it inside a machine, or a product. They might not advertise that IntelligentPad is in use, just as software developers do not ordinarily tell people what language their programs are based on.

SERVICE BUREAUS

Service bureaus, or companies which perform services using software of their own choice or creation, should be an important outlet for IntelligentPad applications.

Market Preparation

Steps must be taken to prepare markets for a product. These have to do with making the public aware-- public relations and magazine placement; with involving opinion leaders-- key user placement; with defining the market and fine-tuning the product (often done with focus groups).

This is the sort of thing that Dentsu ~~would charge a~~ charges million dollars for, and rightly so.

Market Seeding (Business)

Database

Form-based process flow

~~xxx~~

Market Seeding (Non-Business)

A key aspect of getting people to use something is seeding the market. Unix was seeded into universities; Macintoshes were seeded into universities and the graphic arts community.

Steps must be taken to prepare markets for a product. These have to do with making the public aware -- public relations and magazine placement; with involving opinion leaders -- key user placement; with defining the market and fine-tuning the product (often done with focus groups).

This is the sort of thing that Dentsu would charge a million dollars for; and rightly so.

Market Seeding (Business)

Database

Form-based process flow

Market Seeding (Non-Business)

A key aspect of getting people to use something is seeding the market. Unix was seeded into universities; Macintoshes were seeded into universities and the graphic arts community.

THE CHOICES AHEAD = Key Issues

I do not believe that IntelligentPad will sell itself, or have any substantial impact without proper marketing and demo apps.

There have to be choices and they have to be funded.

Market Selection

There must be a choice.

Demo Apps

Something has to be built.
Like the original MacPaint and MacWrite.

KEY DEMO APPS & FRAMEWORKS
Business?
School?
Personal?
Internet?

Market Preparation

There has to be Market Preparation : seeds, publicity.

Channel Preparation

There must be active solicitation of channels.

Rollout

There has to be a Rollout, and it mustn't be botched.

Follow-On

There need to be continuing programs--
DEVELOPER PROGRAM
USER HOT LINE
USER GROUPS

THE CHOICES AHEAD

I do not believe that IntelligentPad will sell itself, or have any substantial impact without proper marketing and demo apps.

There have to be choices and they have to be funded.

Market Selection

Demo Apps

Something has to be built.
Like the original MacPaint and MacWrite.

KEY DEMO APPS & FRAMEWORKS
Business?
School?
Personal?
Internet?

Market Preparation

Channel Preparation

Rollout

There has to be a Rollout, and it mustn't be botched.

Follow-On

There need to be continuing programs—
DEVELOPER PROGRAM
USER HOTLINE
USER GROUPS

Summary and Conclusion

Like many others, I want to see Professor Tanaka achieve his dream. I believe it will take considerable resources and planning. These notes are intended to share my thoughts on the matter.

~~Here~~ I have listed some of the issues we need to address,

But [1] Whatever the resources, they can only begin to cover a small part of the possibilities of IntelligentPad. But I believe a selection should be made and a concerted plan put together.

by why some resources

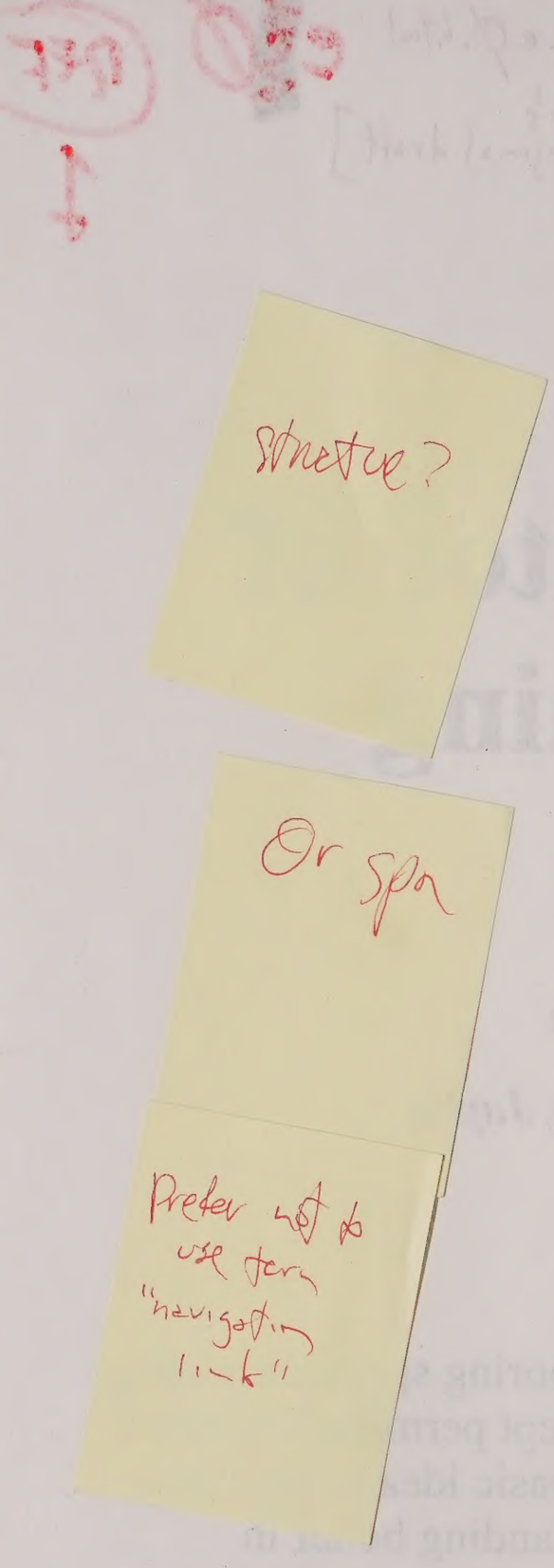
Ziped.w.oo.e∅.html [Mr. Ookubo's original draft]

ZipEditor: A Collaborative Workspace Editor of Shared Versioning Text

Kiyoki Ookubo; HyperLab; Sapporo, Japan.
ookubo@hyper.sec.or.jp
Theodor Holm Nelson; HyperLab; Sapporo, Japan.
ted@hyper.sec.or.jp, ted@xanadu.com

Abstract:

We have developed a collaborative authoring space consisting of text fragments whose identities are kept permanently while allowing free citation and update. The basic idea behind this system came from Ted Nelson's long standing belief in *transclusion*. His naming of hypertext is accepted widely and its accompaing hyperlink concept is widely understood[1]. In spite of this situation, transclusion gains only limited attention[2]. Although our system named *ZipEditor* handles only text, it shows the strength of transclusion as a basic mechanism for editing document which requires long term maintenance and coauthoring. We have also developed WWW connection to give a preliminary experinence how the file manager part of the ZipEditor works.


Structure?
Or spa
Prefer not to use term "navigation link"

Keywords:

transclusion, implied link, side-by-side visualization, versioning, ZipEditor, Web connection

Introduction

We at Sapporo HyperLab have developed the distributed ZipEditor as one part of its research goals. Sapporo HyperLab was established as a short term (1.5 years) collaborative research lab among several Japanese companies. The ZipEditor was intented to develop a transclusion editor using the IntelligentPad System[3] for its user interface. In this paper we will first introduce trasclusion, and then explain how the ZipEditor implements its semantics, and finally discuss its implication to the WWW.

What is transclusion?

Hyperlink is a connection starting from an anchor, like an underlined word or a part of graphic image in the case of the WWW, and ending to the related HTML page. Transclusion means a sameness relation among data objects. A document can be viewed as a thread piercing together document fragments as shown in Figure 1. (The WWW do not provide a way to express this kind of multi-hop thread, but only provides single hop navigation link. The author usually prepares an index page to represent a natural sequence to read). In this example there are two threads and the fragment C is shared among them. This picture can be rearranged like Figure 2.

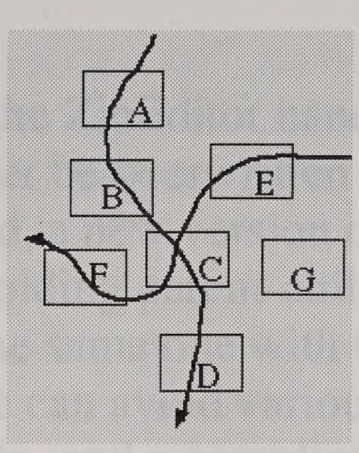


Figure 1: Document fragments and threads pierce together them

The thick line among two C's is what we call transclusion link. So, the transclusion link is not a kind of link the user connects manually. This kind of link is an *implied link* derived from the data structure itself. In Figure 2 the thread in Figure 1 is disappeared and the vertical sequence of fragments takes the role of it. We can say these two representations are equivalent. The ZipEditor adopts this *side-by-side* visualization style and this is convenient for comparing several documents which share some parts, or to compare different versions of derivative work of that document. In the case of different versions of fragment, it can not be represented by a sameness relationship, so that the ZipEitor uses a thin line to visualize the relationship among family document fragments who belong to the same version tree. Transclusion means inclusion and reuse of the same fragment.

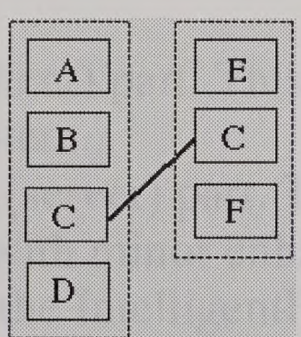


Figure 2: Document threads being rearranged side by side

Figure 1: Document fragments and threads pierce together them

Figure 2: Document threads being rearranged side by side

The data objects which the ZipEditor handles are immutable, i.e. the data objects will never be overwritten nor thrown away. When the data object is updated, a new version of the object is simply created and the record holding parent-children relationship is stored. This attitude is the same one with the lendary Xanadu system had taken and we can avoid various problems, like an obsolete link, a lost origin, and a complicated concurrency control as is stated in [4].

Transclusion has the other important meaning when applied to the open network environment like the Internet. Its tracking capability to the original author can be utilized to solve the copyright problem mentioned in [5] democratically [6][7].

The ZipEditor

The ZipEditor exactly follows the visualization style shown in Figure 2 (see Figure 3). The ZipEditor uses the term *zip* instead of thread and *item* for text fragment. The tranclusion links are colored to deleniate connection even when many lines are needed to draw. The same color is used to border the items which are in transclusive relationship.

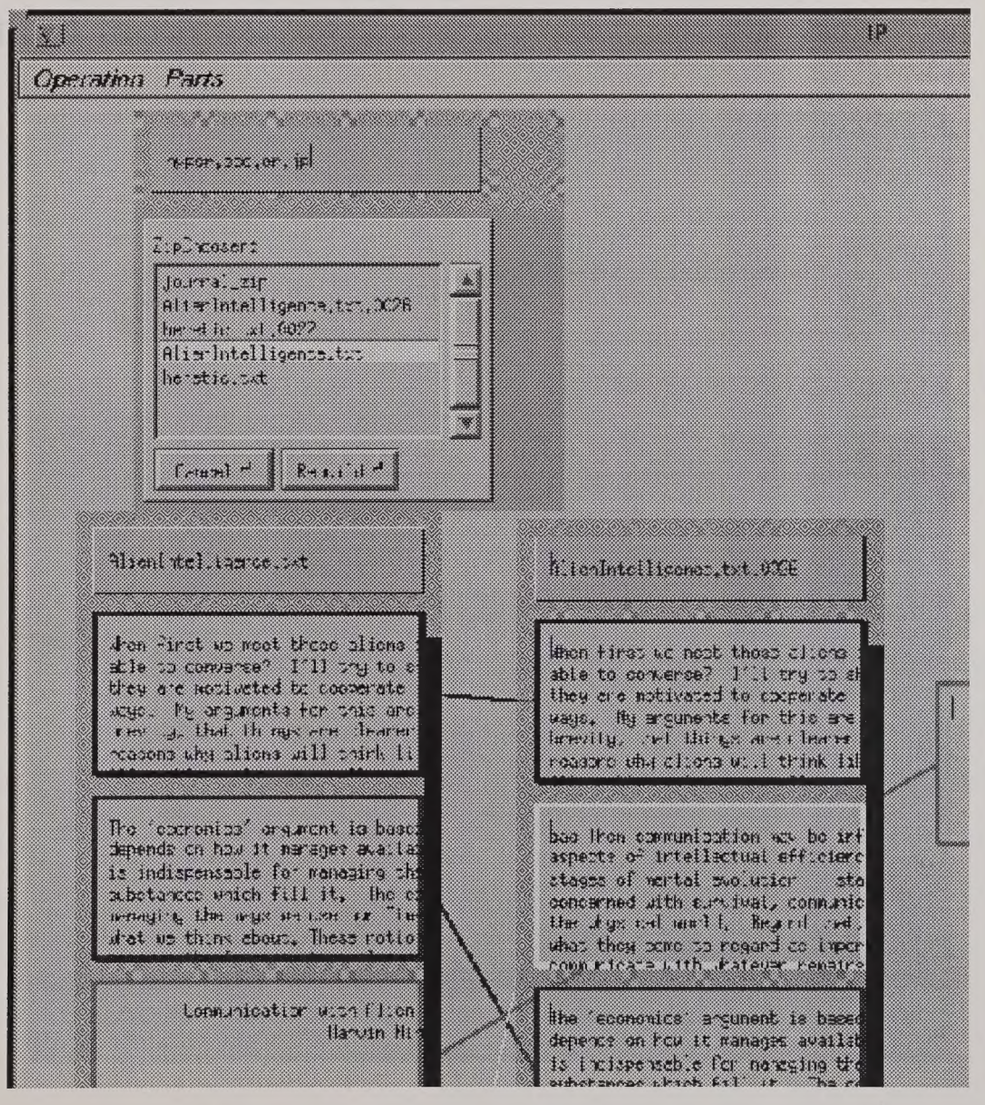
There are two main components in the ZipEditor:

- The user interface (client) part. This is written in c++ using the IntelligentPad System.
- The ZipFileManager (server) part. This is written in Tcl (Tool Comman Language) with minor enhancement. Tcl is also used as a script languge in the IntelligentPad System.

The visible user interface components are all pads. Because of the

pad nature of component independence these visual parts are easy to rearrage. And some pads are translucent to make a lot of zips to be placed on the screen while showing as many information as possible.

The pad positioned in upper part in Figure 3 is called ZipChooser and it has an input form of the server address together with the list contains all zips served by the server. The two zip pads in the lower position are loaded by choosing from the list. Since the ZipChooser can be created a number of as desired, the user can handle zips coming from several different servers. As a result, the user can rearrage items or create new versions of item or zip originated from different servers without almost noticing it (except network delay etc.).



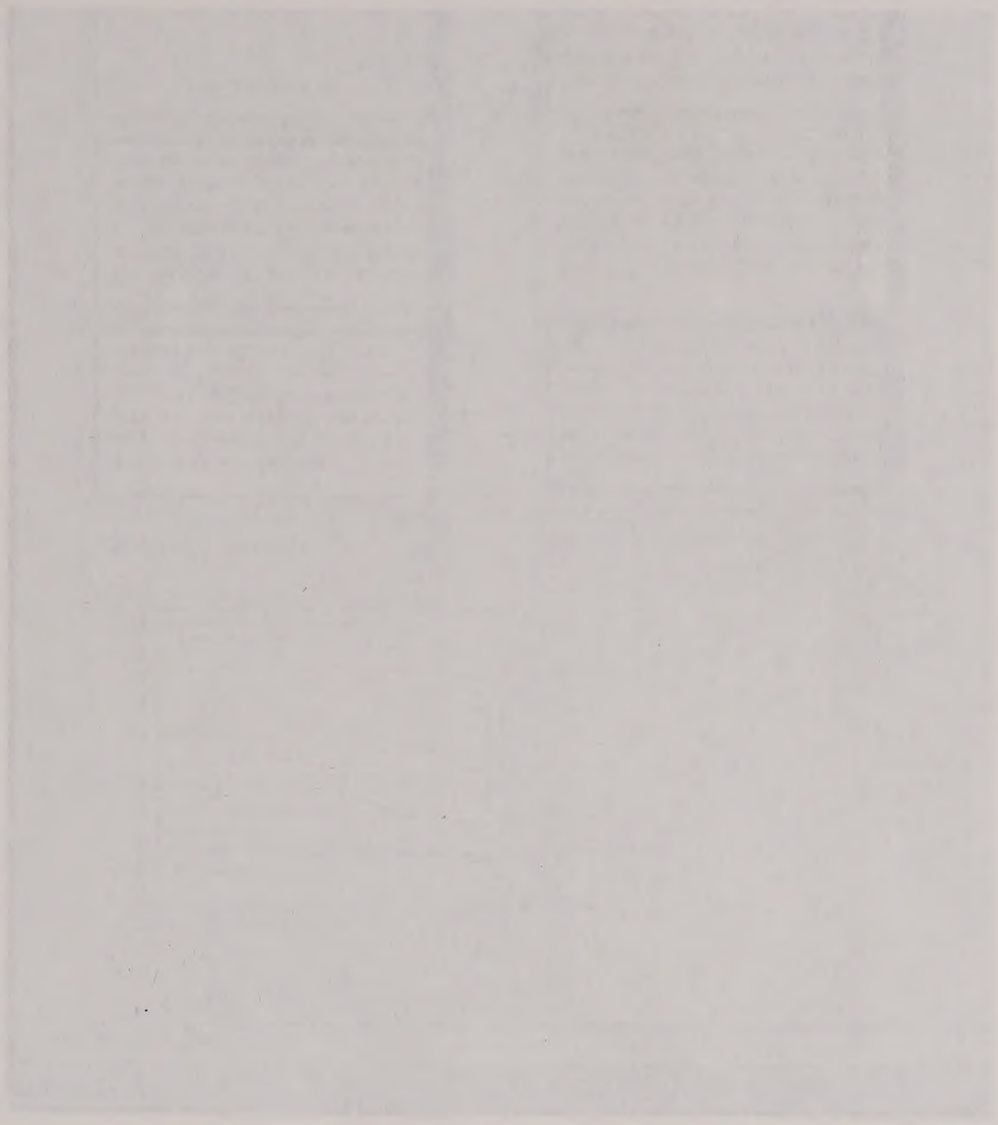
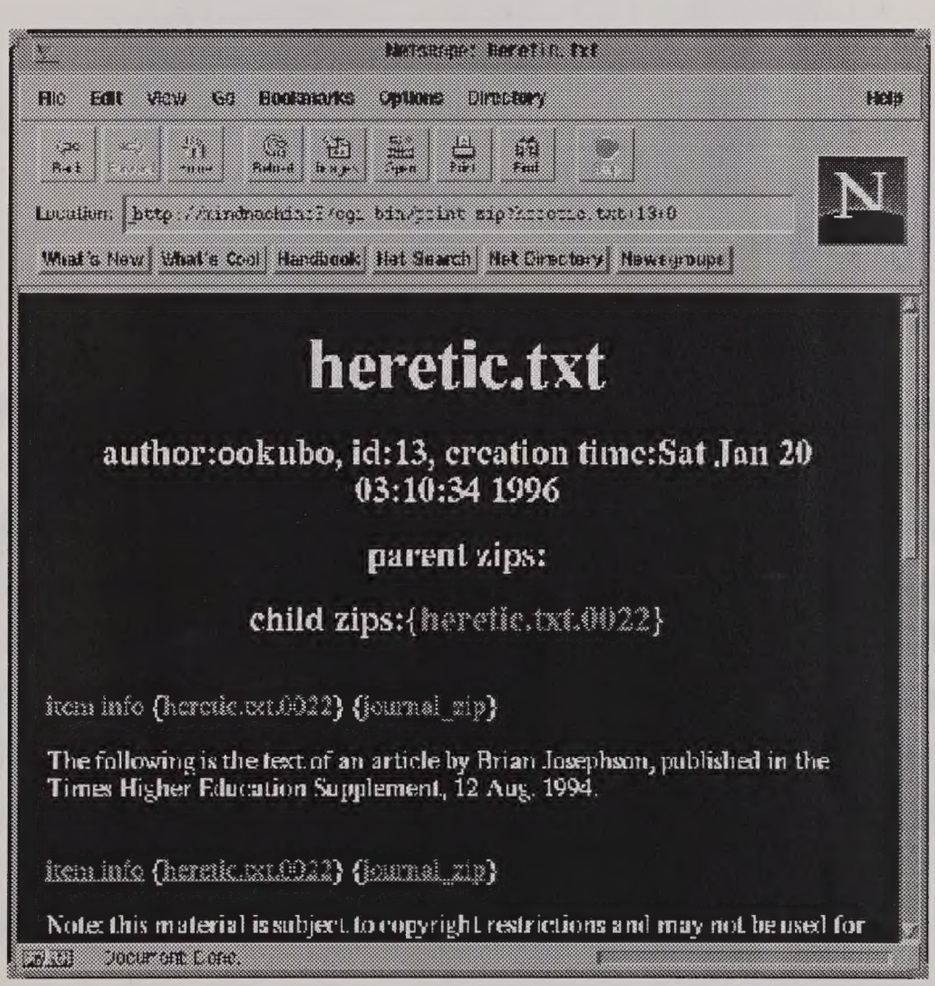


Figure 3: The ZipEditor Screen

The ZipFileManager also supports a capability to respond to the Web server's CGI (Common Gateway Interface) request and generates a HTML page which contains HREF links matching to the transclusions and version relationships. The user can navigate through documents which share parts of the text, or examine who created which parts. When the user uses with the Netscape Browser, he can control the Browser to bring up a Web page corresponding to the item or the zip from the ZipEditor window by selecting menu from there. This is implemented using the openURL of the CCI (Common Client Interface). This kind of usage of the ZipEditor as a remote controller makes the Web hotlist management by a zip representation attractive (not implemented though).


heretic.txt
author:ookubo, id:13, creation time:Sat Jan 20 03:10:34 1996
parent zips:
child zips:{heretic.txt.0022}
item info {heretic.txt.0022} {journal_zip}
The following is the text of an article by Brian Josephson, published in the Times Higher Education Supplement, 12 Aug. 1994.
item info {heretic.txt.0022} {journal_zip}
Note: this material is subject to copyright restrictions and may not be used for

(val
string)

Figure 4: The Web Connection

The Implementation

The overall architecture of the ZipEditor is shown in Figure 5. The interface between the client and the ZipFileServer is a remote call of Tcl interpreter using a string of Tcl command. The user's operation like move, copy, rearrange, delete (from screen), and load etc. generates a corresponding Tcl command and it is sent to the ZipFileServer, and the server updates the database.

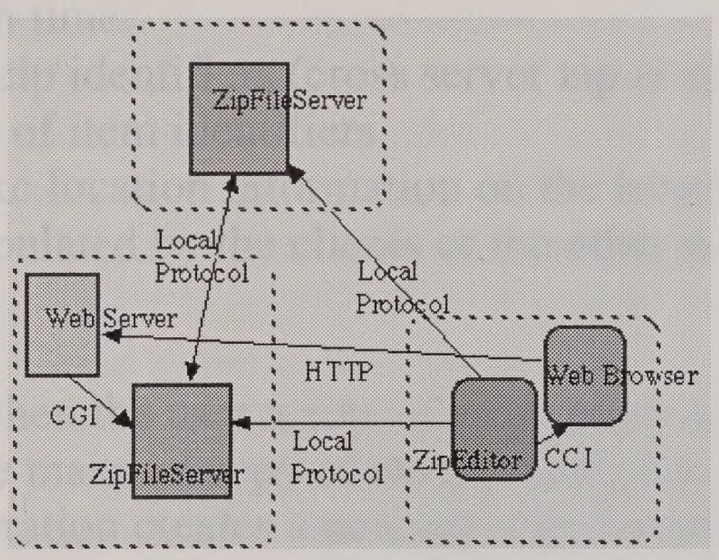


Figure 5: The Architecture of the ZipEditor

The ZipFileServer manages a database containing zips and items, and it maintains their identifications and vesion relationships. Basically a zip contains a sequence of pointers to item. When the user requests to show the content of the zip, the text contained in the item is retrieved for each item by item. The sharing of items among the zips is realized by the pointers to the same item.

The informations of which an item holds are:

sorry about pronouns "he or she"

- the author name.
- the creation time.
- the parent item identifiers (cross server item is allowed).
- the identifier of the zip on which it was born.
- a text fragment.
- an URL-like location information on the Internet when the item is manipulated by the clients or the other servers than it belongs to.

The informations of which a zip holds are:

- the author name.
- the creation time.
- the parent zip identifiers (cross server zip is allowed).
- a sequence of item identifiers.
- an URL-like location information on the Internet. when the zip is manipulated by the clients or the other servers than it belongs to.

For example, when the user wants to reuse some zip written by the other person, he makes a copy of the zip using the ZipEditor client. This operation creates a new zip object containing the same sequence of item identifiers with the original zip, but with the new author name and contains the original zip identifier as its parent. And then the user modifies text in some item using the ZipEditor client, a new item object containg the modified text and the pointer to the original item is created. At the same time the element standing for the original item in the zip is replaced by the newly created item identifier. In our current implementation, the replacement of item in the zip causes a direct modification of the zip object and no information is kept to revert to the previous state. The recording of operation is necessary to maintain the logical immutability.

There are many things left in our wish list. Among these things, we want to experiment the next two things in our research term.

- a simple accounting mechanism for making payment to the original author possible. Because this is the other side of transclusion.
- an introduction of digital signature technique for guaranteeing free flow of information still preventing tampering against it.

Conclusion

The success of the WWW has increased the information available to us dramatically and it makes our net life more exciting. To continue healthy progress, we consider the mechanism presented here need to be incorporated into the WWW in someway or the other. Although people want to have an inline text, pepole knows it cause problem. There is another reason to motivate adoption of transclusion. It relates to the area known as representation of knowledge by the Web. We expect this effort will change the Web scape more structured one. Transclusion is a fundamental concept for sharing information fairly and the first citizen of implied links which the knowledge representation needs to have.

Acknowledgement:

Prof. Yuzuru Tanaka of Hoddaido Univ. is the inventor of the IntelligentPad system and participated the discussion of the architecture of the ZipEditor. Tatsuya Kainuma of Fujitsu Aichi Engineering, Inc. implemented a version of the IntelligentPad system used in the ZidEditor and he taught us how to use it. The stuffs of Sapporo Electronic Center Foundation provided the research environment.

There are many things left in our wish list. Among these things, we want to experiment the next two things in our research term.

- a simple accounting mechanism for making payment to the original author possible. Because this is the other side of transclusion.
- an introduction of digital signature technique for guaranteeing free flow of information still preventing tampering against it.

Conclusion

The success of the WWW has increased the information available to us dramatically, and it makes our net life more exciting. To continue healthy progress, we consider the mechanism presented here need to be incorporated into the WWW in someway or the other. Although people want to have an inline text, people knows it cause problem. There is another reason to motivate adoption of transclusion. It relates to the area known as representation of knowledge by the Web. We expect this effort will change the Web's shape more structured one. Transclusion is a fundamental concept for sharing information fairly and the first citizen of implied links which the knowledge representation needs to have.

Acknowledgements

Prof. Yuzuru Tanaka of Hokkaido Univ. is the inventor of the IntelligenPad system and participated the discussion of the architecture of the ZipEditor. Tatsuya Kamiura of Fujitsu Aichi Engineering, Inc. implemented a version of the IntelligenPad system used in the ZipEditor and he taught us how to use it. The staffs of Sapporo Electronic Center Foundation provided the research environment.

References

[1] Searching *hyperlink* from Alta Vista, about 10000 matching was found on January 24, 1996.

[2] Searching *transclusion* from Alta Vista, 79 matching was found on January 24, 1996.

[3] The IntelligentPad System Page of Hokkaido University

[4] Fabio Vitali, David G. Durand. "Using versioning to support collaboration on the WWW", *Proceedings of the Fourth International WWW Conference '95, Boston, Massachusetts, USA, December 11-14, 1995.*

[5] Terje Norderhaug, Juliet M. Oberding. "Designing a Web of Intellectual Property", *Proceedings of the Third International WWW Conference '95, Darmstadt, Germany, April 10-14, 1995.*

[6] Theodor Holm Nelson "A Publishing and Royalty Model for Networked Documents", *Proceedings: Technological Strategies for Protecting Intellectual Property in the Networked Multimedia Environment, 1995.*

[7] Theodor Holm Nelson "Transcopyright: Pre-Permission for Virtual Republishing", *Date of this version: 95.4.9.*

```
# makezip.e3
# 95.7.23


# A cell's name will be "cellN", eg cell1, cell276
# We'll keep the current cell name in "currentcellname"

# The cell's type will be in cellN(type).

#  Eventually,
#  The cell's contents will be indicated by a pointer pair stored in
#  cellN(contentptrs)
#   HOWEVER, for debugging purposes we'll have a variable called
#  cellN(contents)

# We could even have a cell NAME, in cellN(name)!

# Convention for dimension subscripts: always use plus and minus,
#  so we'll know that "stack" in
#  cellN(+stack) is a dimension, for instance.


# SO LET'S MAKE A CELL:

# ------------------------- THIS WORKS FINE:



#set cell0(type) unknown
#set cell0(contents) "Hi there!  I'm the contents!"
#set cell0(name) HOME
#set cell0(+t) cell1


#set cell1(type) unknown
#set cell1(contents) "Hi there!  I'm the contents!"
#set cell1(-t) cell0



# ----------------------- THIS DOESN'T WORK:
# -- how keep changing name of array?

#set curno 7
# set cell$curno(type) experimental
# set cell$curno(-1) cell24
#    The error message is: invalid command name "7"

#------------------------- THIS DOESN'T WORK EITHER

# set a [ join cell [ list $curno]]


#------------------------- LET'S DO IT AS A HACK--
#First, to legitimize the array--


set a "set "

set cellname cell23

set b "(dummy) 3"

set legitcommand $a$cellname$b

eval $legitcommand

# this should legitimize the array.
```

```
# cellspecs.e4
#  from  makezip.e3
# 95.7.23

# A cell's name will be "cellN", eg cell1, cell276
# We'll keep the current cell name in "currentcellname"

# The cell's type will be in cellN(type).

#  Eventually,
#  The cell's contents will be indicated by a pointer pair stored in
#  cellN(contentptrs)
#   HOWEVER, for debugging purposes we'll have a variable called
#  cellN(contents)

# We could even have a cell NAME, in cellN(name)!

# Convention for dimension subscripts: always use plus and minus,
#  so we'll know that "stack" in
#  cellN(+stack) is a dimension, for instance.
```

```
# makezip.e2
# 95.7.23

# A cell's name will be "cellN", eg cell1, cell276
# We'll keep the current cell name in "currentcellname"


# The cell's type will be in cellN(type).


#  The cell's contents will be indicated by a pointer pair stored in
#  cellN(contptr)
#   HOWEVER, for debugging purposes we'll have a variable called
#  cellN(contents)

# We could even have a cell NAME, in cellN(name)!

# Convention for dimension subscripts: always use plus and minus,
#  so we'll know that cellN(+stack) is a dimension, for instance.


# SO LET'S MAKE A CELL:

# ------------------------ THIS WORKS FINE:


set cell0(type) unknown
set cell0(contents) "Hi there!  I'm the contents!"
set cell0(name) HOME
set cell0(+t) cell1

set cell1(type) unknown
set cell1(contents) "Hi there!  I'm the contents!"
set cell1(-t) cell0


# ------------------------ THIS DOESN'T WORK:
# -- how keep changing name of array?

# set curno 7

# set cell[$curno](type) experimental

# set cell[$curno](-1) cell24

#    The error message is: invalid command name "7"


#------------------------- THIS DOESN'T WORK EITHER

# set a [ join cell [ list $curno]]
```

```
# makezip.el
# 95.7.23

# A cell's name will be "cellN", eg cell1, cell276
# We'll keep the current cell name in "currentcellname"


# The cell's type will be in cellN(type).


#  The cell's contents will be indicated by a pointer pair stored in
#  cellN(contptr)
#   HOWEVER, for debugging purposes we'll have a variable called
#  cellN(contents)


# SO LET'S MAKE A CELL:

# ------------------------ THIS WORKS FINE:

set curno 23

set cell23(type) experimental


# ----------------------- THIS DOESN'T WORK:

# set curno 7

# set cell[$curno](type) experimental

# set cell[$curno](-1) cell24

#    The error message is: invalid command name "7"


#------------------------- THIS DOESN'T WORK EITHER

# set a [ join cell [ list $curno]]
```

```
# legit.array.works
# 95.7.23

#(from makezip.e3)

# A cell's name will be "cellN", eg cell1, cell276
# We'll keep the current cell name in "currentcellname"


# The cell's type will be in cellN(type).


#  Eventually,
#  The cell's contents will be indicated by a pointer pair stored in
#  cellN(contentptrs)
#   HOWEVER, for debugging purposes we'll have a variable called
#  cellN(contents)

# We could even have a cell NAME, in cellN(name)!

# Convention for dimension subscripts: always use plus and minus,
#  so we'll know that "stack" in
#  cellN(+stack) is a dimension, for instance.


# SO LET'S MAKE A CELL:

# ------------------------- THIS WORKS FINE:



#set cell0(type) unknown
#set cell0(contents) "Hi there!  I'm the contents!"
#set cell0(name) HOME
#set cell0(+t) cell1


#set cell1(type) unknown
#set cell1(contents) "Hi there!  I'm the contents!"
#set cell1(-t) cell0



# ------------------------ THIS DOESN'T WORK:
# -- how keep changing name of array?

#set curno 7
# set cell$curno(type) experimental
# set cell$curno(-1) cell24
#    The error message is: invalid command name "7"

#------------------------- THIS DOESN'T WORK EITHER

# set a [ join cell [ list $curno]]


#------------------------- LET'S DO IT AS A HACK--
#First, to legitimize the array--


set a "set "

set cellname cell23

set b "(dummy) 3"

set legitcommand $a$cellname$b

eval $legitcommand

# this should legitimize the array.

# ------------------------- NOW CAN WE SUBSCRIPT FREELY?

# THIS STILL DOESN'T WORK--

# set $cellname(+1) 3

# --to which the error message is:
```

can't read "cellname(+1)": variable isn't array

But I didn't try to read cellname(+1), I tried to read-- Oh.